MPS_board : Board which generate an interface between the primary data acquisition Crate and the French Data Acquisition crate for the MPS signal in the G0 experiment (included a MPS signal generator test)

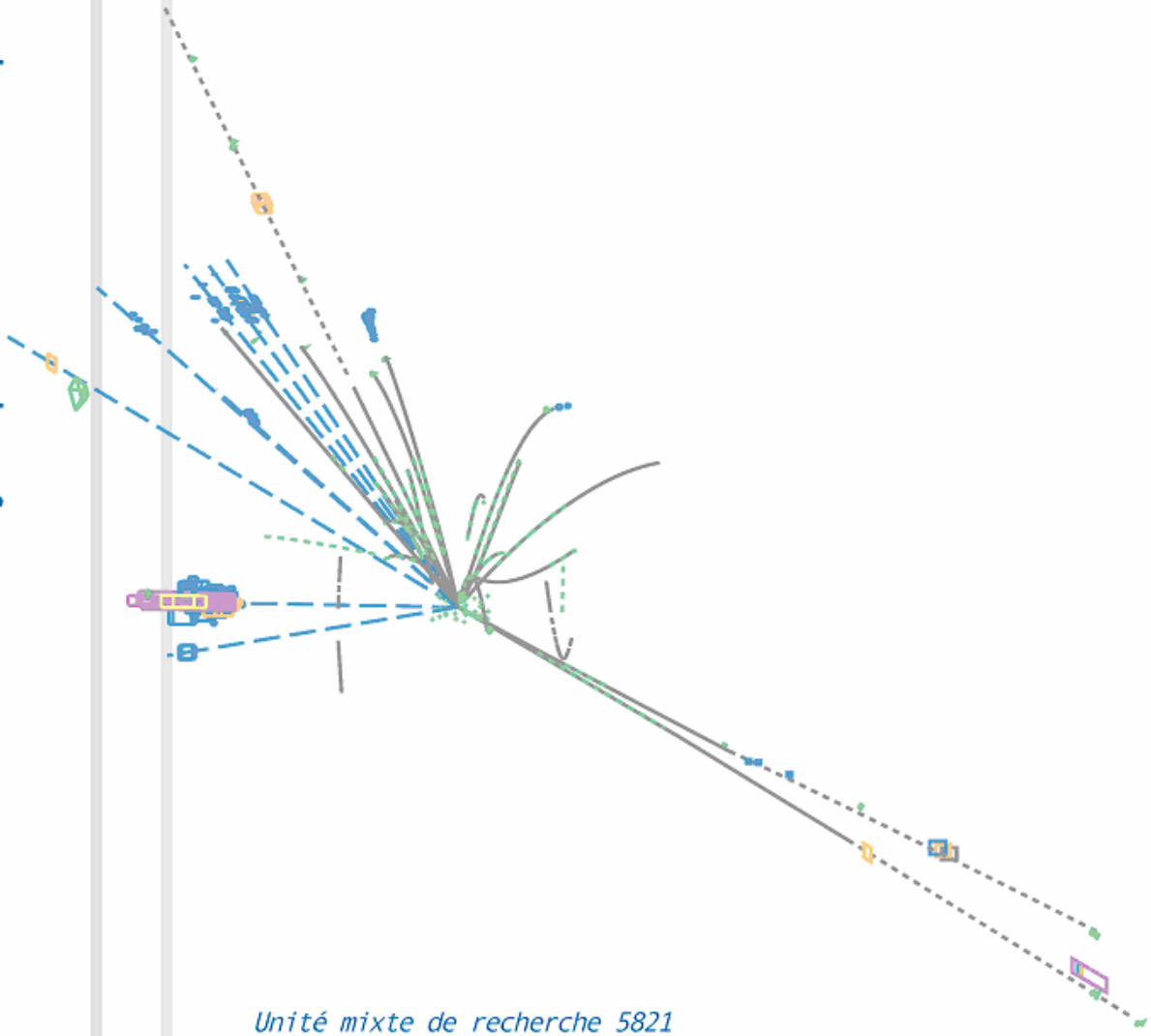
J. Bouvier

Internal report LPSC 05-91
Ver.: 1
September 2005

MPS_BOARD

Board which generate an interface between the primary data acquisition Crate and the French Data Acquisition crate for the MPS signal in the G0 Experiment (included a MPS signal generator test)

Joel BOUVIER
Data Acquisition Team



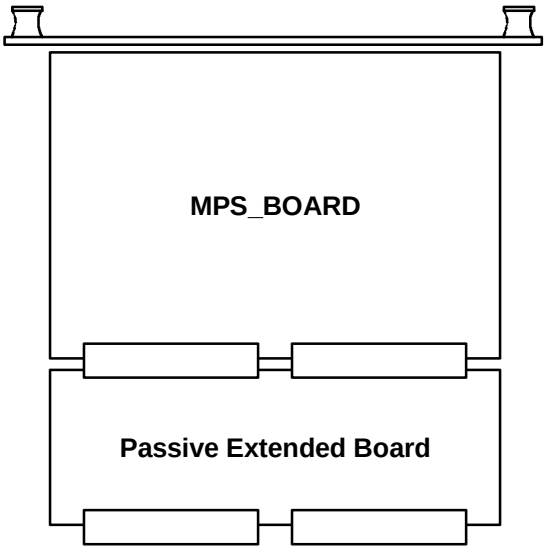
Laboratoire de Physique Subatomique et de Cosmologie
Unité mixte de recherche 5821
CNRS-IN2P3 / UJF / INPG

1. Overview

The goal of this board is to have the possibility to select which signal is present on the MPS line on the VXI Subsystem slot.
This selection is done by the Data Acquisition software and the number of signal source is 4. Two of them are present on the front end board and the 2 others are internal board signals.

2. Functionalities

The board is a four to one signal multiplexer with enable and invert output function.
Five front end board indicator (LED) shows which source is selected (if one is selected) and the actual state of the INVERT SIGNAL command.
The board is a two layers printed board without plated-through hole with a VXI B size mechanical dimension.
To put this board in a VXI C size mechanical crate a special passive extended board is need.



2.1. Signal source

The four input signal source are :

- ✓ IN_RJ45 input
- ✓ IN_LEMO input
- ✓ 120Hz internal board generator
- ✓ 30Hz internal board generator

Each of them is selected by an internal board register where all sources are defined by a bit plus one bit for the INVERT SIGNAL command.

2.1.1. IN_RJ45 input

This signal present on front end board is a RS422 input signal level compatible packaged in a RJ45 type connector (like ETHERNET connector).
This signal can be generated by a BPO_EXT board previously developed for this experiment. A simple Ethernet board can connected the two boards.

Tableau 1 : RJ45 Connector pinout

Pin	Name
1	N.C.
2	N.C.
3	N.C.
4	N.C.
5	Gnd
6	Gnd
7	IN-
8	IN+

2.1.2. IN_LEMO input

This input signal packaged by a LEMO type connector is a multi level compatible input.
This input is terminated by a 50 ohm resistor.
The hardware input is like the following figure which shows how the level is chosen.

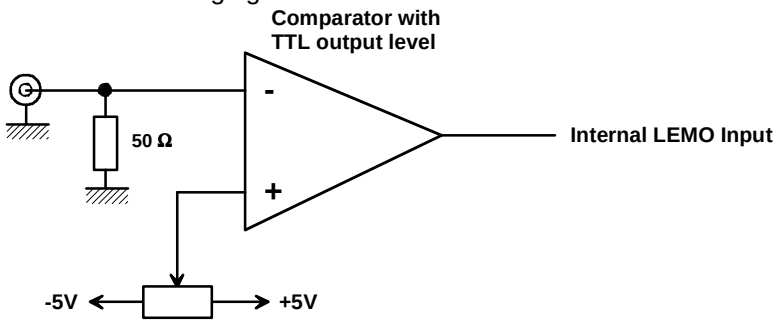


Figure 1 : LEMO Input description

The potentiometer permit to choose the commutation level (from -5v to +5v).
WARNING : for the positive commutation level, the output give an inverted signal. To make a output board signal with a right phase the programmer must used the inverted output command.

2.1.3. 120Hz internal board generator

The MPS_BOARD have the possibility to generate an 120Hz signal like the following figure. This 120Hz signal is like the 120Hz signal generated by the beam generator. This signal serves as a test when the COINCIDENCE crate function in a stand alone mode.
When this option is selected the first state generated is the inactive state (level 1 during 3 µs) and after the active state is generated (level 0 during 8,331 ms) and so on.
When a writing access from the VXI backplane is done , access enabled this internal generator, the beginning state is always the same like the sequence describe in the previous paragraph.

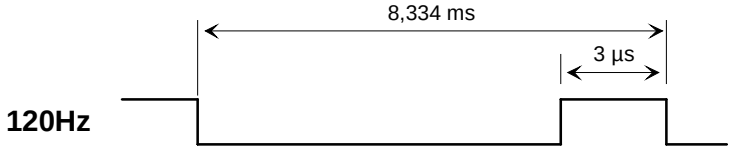


Figure 2 : 120Hz internal generator waveform

2.1.4. 30Hz internal board generator

The MPS_BOARD have the possibility to generate an 30Hz signal like the following figure. This 30Hz signal is like the 300Hz signal generated by the beam generator. This signal serves as a test when the COINCIDENCE crate function in a stand alone mode.

When this option is selected the first state generated is the inactive state (level 1 during 200 µs) and after the active state is generated (level 0 during 33,2 ms) and so on.
When a writing access from the VXI backplane is done , access enabled this internal generator, the beginning state is always the same like the sequence describe in the previous paragraph.

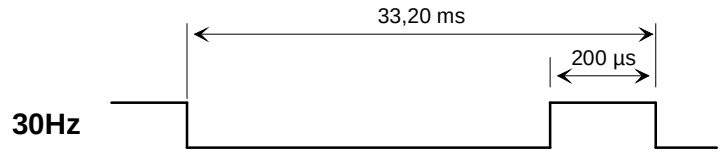


Figure 3 : 30Hz internal generator waveform

2.2. Output board signal (MPS signal on the backboard)

The signal present on this line is conform to the selected signal thought the internal board command register describe in the following chapter.
If no signal is selected this line is at a high level logic (inactive state).

2.3. Internal board command register

The board have an unique register. This register is read or written thought all the board space address (64K bytes).

The board is see thought the following VXI address modifier :

- ✓ Standard supervisory Ascending Access : x”3F”
- ✓ Standard non-privileged ascending access : x”3B”
- ✓ Standard supervisory Program Access : x”3E"
- ✓ Standard non-privileged Program access : x”3A”
- ✓ Standard supervisory data Access : x"3D"
- ✓ Standard non-privileged data access : x"39"
- ✓ Extended supervisory Ascending Access : x"0F"
- ✓ Extended non-privileged ascending access : x"0B"
- ✓ Extended supervisory Program Access : x”0E"
- ✓ Extended non-privileged Program access : x”0A”
- ✓ Extended supervisory data Access : x"0D"
- ✓ Extended non-privileged data access : x"09"

x” “ represents an hexadecimal value.

2.3.1. internal board command register description

The internal board command register is like the following description :

	D[31..8]	D7	D6	D5	D4	D3	D2	D1	D0
Base_adress					30Hz	120Hz	IN_LEMO	IN_422	INVERT

All the bits is set to 0 by a power up or a RESET

D0 : INVERT when this bit is set to one the MPS output signal is inverted compared to the input signals or to the internal generators signals. The default value of this bit is ‘0’.

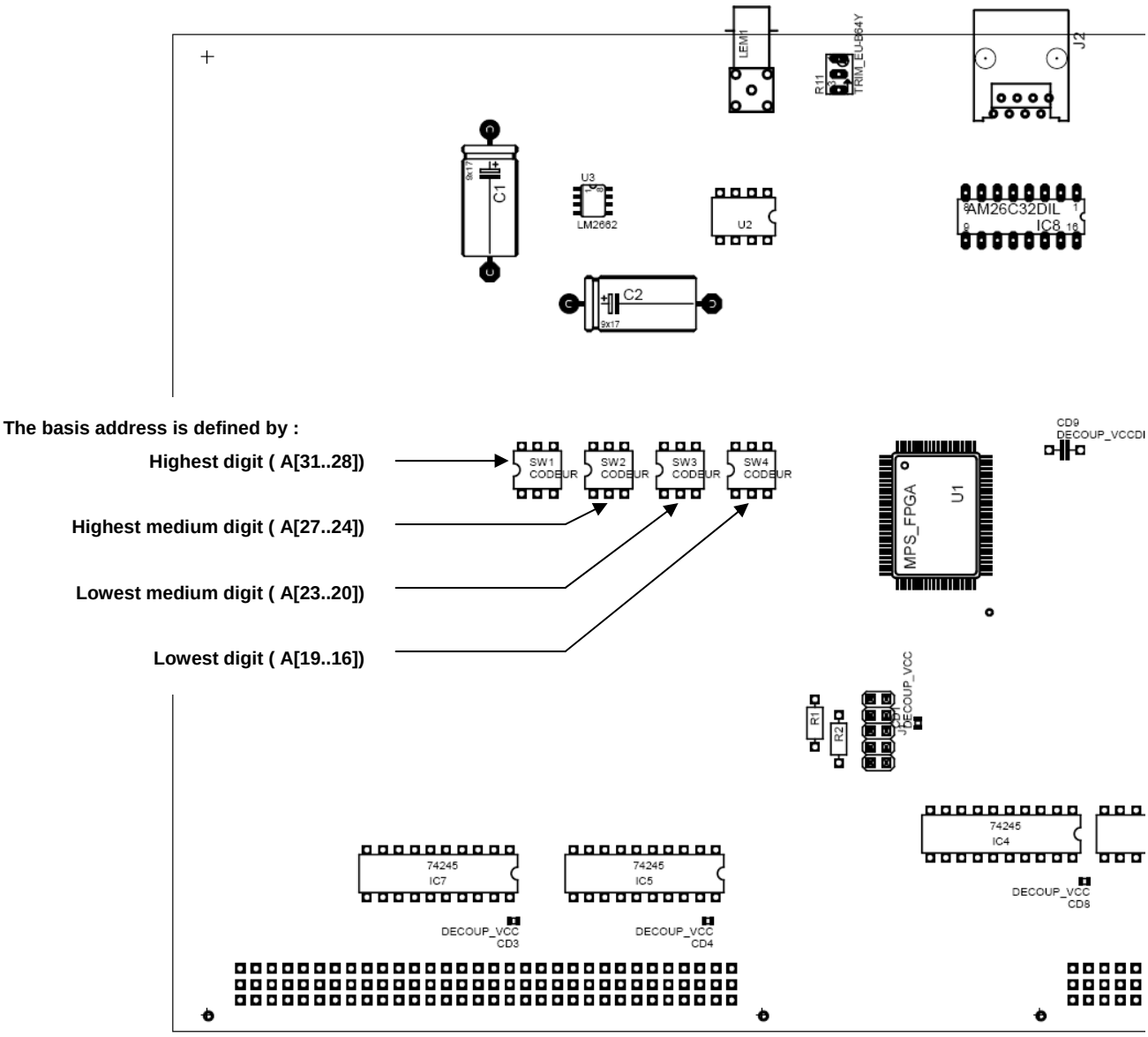
D1 : IN_422 when this bit is set to one the signal present on the RS45 connector is transferred to the MPS output signal

D2 : IN_LEMO when this bit is set to one the signal present on the LEMO connector is transferred to the MPS output signal

D3 : 120Hz when this bit is set to one the output signal of the FPGA internal 120Hz generator is transferred to the MPS output signal

D4 : 30Hz when this bit is set to one the output signal of the FPGA internal 30Hz generator is transferred to the MPS output signal

2.3.2. base address selection



Annex 1 : Pin out of the VXI connector

P1 Connector

N°	Row A	Row B	Row C
1	D00	/BBUSY	D08
2	D01	/BCLR	D09
3	D02	/ACFAIL	D10
4	D03	/BG0IN	D11
5	D04	/BG0OUT	D12
6	D05	/BG1IN	D13
7	D06	/BG1OUT	D14
8	D07	/BG2IN	D15
9	GND	/BG2OUT	GND
10	SYSCLK	/BG3IN	/SYSFAIL
11	GND	/BG3OUT	/BERR
12	/DS1	/BR0	/SYSRESET
13	/DS0	/BR1	/LWORD
14	/WRITE	/BR2	AM5
15	GND	/BR3	A23
16	/DTACK	AM0	A22
17	GND	AM1	A21
18	/AS	AM2	A20
19	GND	AM3	A19
20	/IACK	GND	A18
21	/IACKIN	SERCLK	A17
22	/IACKOUT	SERDAT	A16
23	AM4	GND	A15
24	A07	/IRQ7	A14
25	A06	/IRQ6	A13
26	A05	/IRQ5	A12
27	A04	/IRQ4	A11
28	A03	/IRQ3	A10
29	A02	/IRQ2	A09
30	A01	/IRQ1	A08
31	- 12	+ 5 v STDBY	+ 12 v
32	+ 5 v	+ 5 v	+ 5 v

P2 Connector

N°	Row A	Row B	Row C
1	HF	+ 5 v	CLK10+
2	- 2 V	Gnd	CLK10-
3	/HF	Reserved	Gnd
4	Gnd	A[24]	- 5,2 V
5		A[25]	
6		A[26]	
7	- 5,2 V	A[27]	Gnd
8		A[28]	
9		A[29]	
10	Gnd	A[30]	Gnd
11		A[31]	
12		Gnd	
13	- 5,2 V	+ 5 v	- 2 V
14		D[16]	
15		D[17]	
16	Gnd	D[18]	Gnd
17		D[19]	
18	HELin (local bus 09)	D[20]	HELout (local bus 09)
19	- 5,2 V	D[21]	- 5,2 V
20	QRTin (local bus 10)	D[22]	QRTout (local bus 10)
21	Localin (local bus 11)	D[23]	Localout (local bus 11)
22	Gnd	GND	Gnd
23	/MPS	D[24]	120H0
24	120H1	D[25]	T120
25	+ 5 V	D[26]	Gnd
26	/MRUN	D[27]	DHF
27	/MRST*	D[28]	LTPO
28	Gnd	D[29]	Gnd
29		D[30]	
30	MODID	D[31]	Gnd
31	Gnd	GND	+ 24 V
32	TST	+ 5 v	- 24 V

Annex 2 : Used Pin on VXI connector

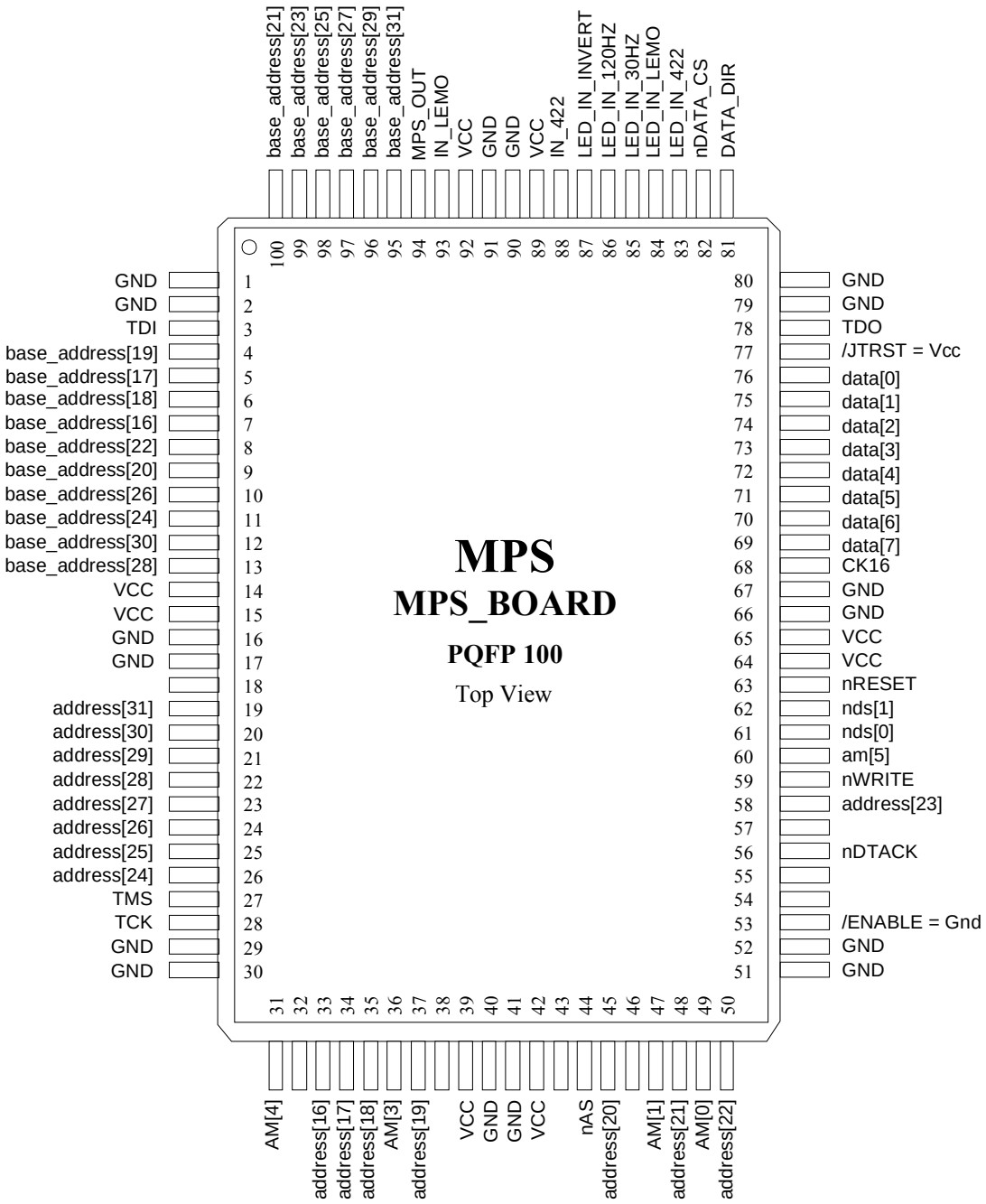
P1 Connector

N°	Row A	Row B	Row C
1	D00		
2	D01		
3	D02		
4	D03		
5	D04		
6	D05		
7	D06		
8	D07		
9	GND		GND
10	SYSCLK		
11	GND		
12	/DS1		/SYSRESET
13	/DS0		
14	/WRITE		AM5
15	GND		A23
16	/DTACK	AM0	A22
17	GND	AM1	A21
18	/AS		A20
19	GND	AM3	A19
20		GND	A18
21			A17
22			A16
23	AM4	GND	
24			
25			
26			
27			
28			
29			
30			
31			
32	+ 5 v	+ 5 v	+ 5 v

P2 Connector

N°	Row A	Row B	Row C
1		+ 5 v	
2		Gnd	
3			
4		A[24]	
5		A[25]	
6		A[26]	
7		A[27]	
8		A[28]	
9		A[29]	
10		A[30]	
11		A[31]	
12		Gnd	
13		+ 5 v	
14			
15			
16			
17			
18			
19			
20			
21			
22			
23	/MPS		
24			
25			
26			
27			
28			
29			
30			
31		GND	
32		+ 5 v	

Annex 3 : FPGA Pin out



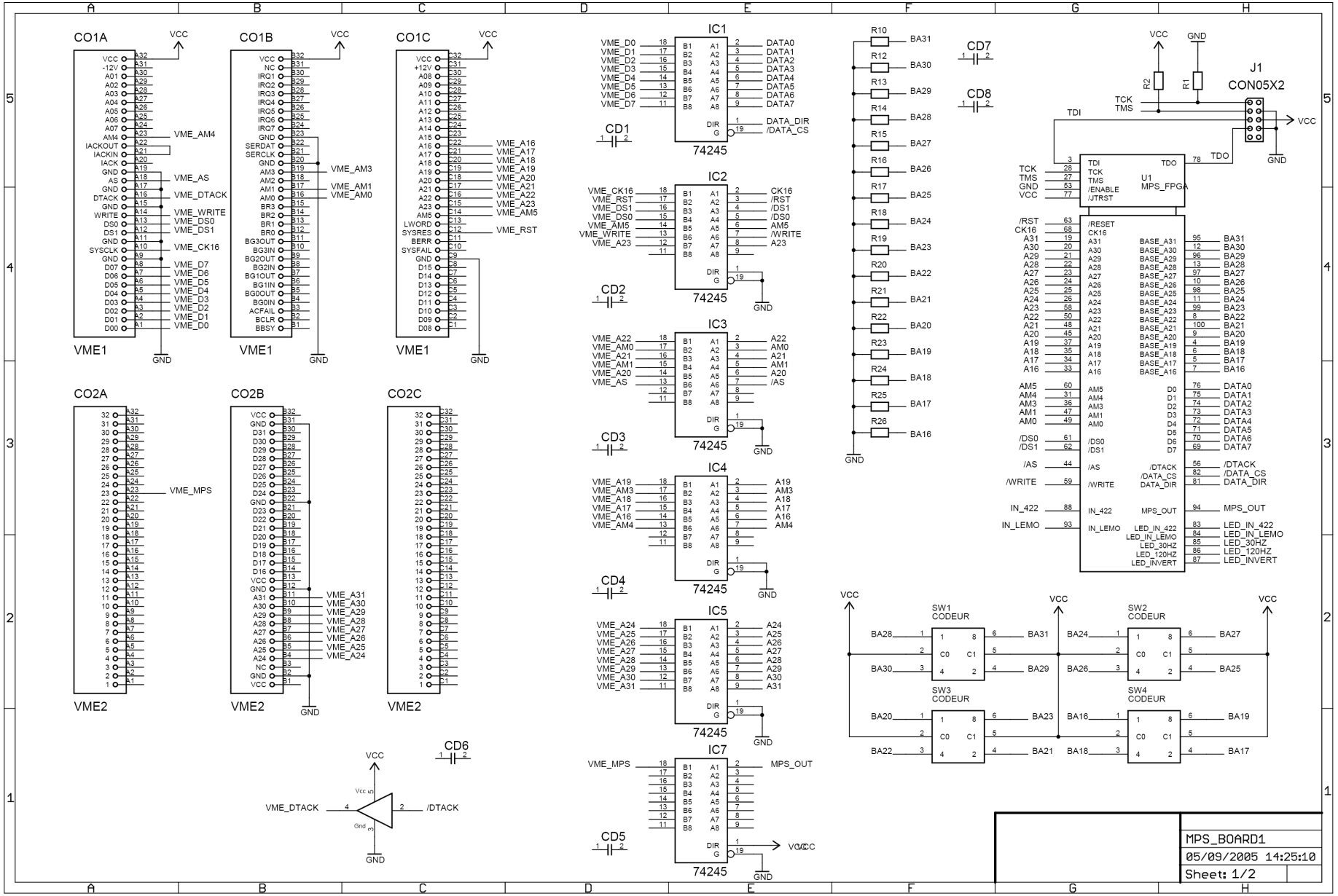
Annex 4 : FPGA VHDL program

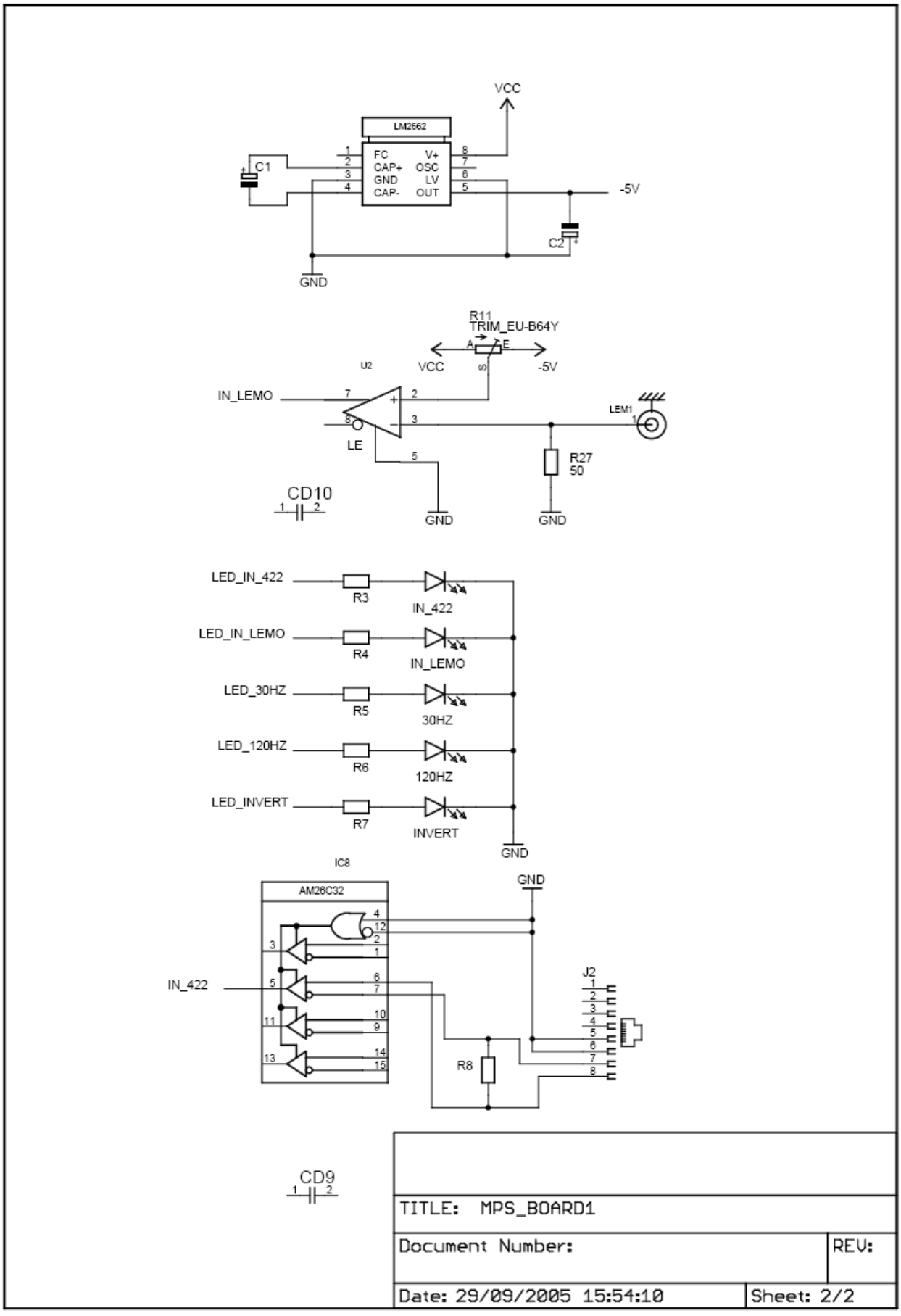
```
--=============================================================================
-- Design Units : MPS Control Board ( Parts of G0 experiment, French parts )
--
-- File name    : mps.vhd
--
-- Purpose      : The purpose of this design is have the possibility to
--                determine which signal is used as FDP MPS signal for the
--                French DAQ in the G0 experiment
--
-- Notes        : the buses is writed in lower-case letter
--                the simple signal is in upper-case letter
--
-- Limitations  :
--
-- Errors       :
--
-- Library      :
--
-- Dependencies :
--
-- Author       : Joel BOUVIER
--                Laboratoire de physique Subatomique et de cosmologie
--                53 Avenue des Martyrs
--                38026 Grenoble Cedex, FRANCE
--
--
--=============================================================================
-- Revision List
-- Version  Author  Date      Change
--    1      J.B.   31/08/05  Initial version
--
--=============================================================================

library ieee;
use ieee.std_logic_1164.all;
use ieee.std_logic_arith.all;
use ieee.std_logic_unsigned.all;

entity mps_board is
  port (
        nRESET       : in    std_logic;
        CK16         : in    std_logic;
        base_address : in    std_logic_vector(31 downto 16);
        address      : in    std_logic_vector(31 downto 16);
        am           : in    std_logic_vector(5 downto 0);
        nds          : in    std_logic_vector(1 downto 0);
        nAS          : in    std_logic;
        nWRITE       : in    std_logic;
        data         : inout std_logic_vector(7 downto 0);
        nDTACK       : out   std_logic;
        nDATA_CS     : out   std_logic;
        DATA_DIR     : out   std_logic;

        IN_422       : in    std_logic;
        IN_LEMO      : in    std_logic;
        LED_IN_422   : out   std_logic;
        LED_IN_LEMO  : out   std_logic;
        LED_30HZ     : out   std_logic;
        LED_120HZ    : out   std_logic;
        LED_INVERT   : out   std_logic;

        MPS_OUT      : out   std_logic );
  end mps_board;

architecture ARCH_carte_gene of mps_board is

  signal   adr_modif       : std_logic_vector(7 downto 0);
  signal   cp_add          : std_logic_vector(1 downto 0);
  signal   dtack_int       : std_logic_vector(2 downto 0);
  signal   ECRIT           : std_logic;
  signal   int_reg         : std_logic_vector(7 downto 0);
  signal   VALID23         : std_logic;
  signal   VALID32         : std_logic;
  signal   MPS_INT         : std_logic;
  signal   INVERT          : std_logic;
  constant BASE_CYCLE_30HZ  : std_logic_vector(11 downto 0) := x"C7F";
  constant LENGTH_30HZ      : std_logic_vector(11 downto 0) := x"0A5";
  constant BASE_CYCLE_120HZ : std_logic_vector(11 downto 0) := x"02F";
  constant LENGTH_120HZ     : std_logic_vector(11 downto 0) := x"AD9";
  signal   cpt_cycle       : std_logic_vector(11 downto 0);
  signal   carry_cycle     : std_logic;
  signal   cpt_length      : std_logic_vector(11 downto 0);
  signal   MPS_INT_GEN     : std_logic;

begin

-----------------------------------------------------------------
--          VME BUS CONTROLER
-----------------------------------------------------------------

adr_modif  <= ('0','0',am(5),am(4),am(3),am(2),am(1),am(0));

--***   Comparaison du bus adresse poid fort aves les switchs de decodage
```

```
cp_add(0) <= '1' when (address(31 downto 24) = base_address(31 downto 24)) else '0';

--***   Comparaison du bus adresse poid fort aves les switchs de decodage

cp_add(1) <= '1' when (address(23 downto 16) = base_address(23 downto 16)) else '0';

--***       validation de l'adresse en mode normal ( A0..A23 )
--***  Seul les acces suivant sont valides
--***     - acces descendant en mode Superviseur & utilisateur standard
--***     - acces au programme en mode Superviseur & utilisateur standard
--***     - acces aux donnees en mode Superviseur & utilisateur standard

valid23 <= '1' when (cp_add(1) ='1' and nAS ='0' and (adr_modif= x"3B" or adr_modif = x"3F")) else
         '1' when (cp_add(1) ='1' and nAS ='0' and (adr_modif= x"3A" or adr_modif = x"3E")) else
         '1' when (cp_add(1) ='1' and nAS ='0' and (adr_modif= x"39" or adr_modif = x"3D")) else
         '0';

--***       validation de l'adresse en mode etendu ( A0..A31 )
--***  Seul les acces suivant sont valides
--***     - acces descendant en mode Superviseur & utilisateur etendu
--***     - acces au programme en mode Superviseur & utilisateur etendu
--***     - acces aux donnees en mode Superviseur & utilisateur etendu

valid32 <= '1' when (cp_add = "11" and nAS ='0' and (adr_modif= x"0B" or adr_modif = x"0F")) else
         '1' when (cp_add = "11" and nAS ='0' and (adr_modif= x"0A" or adr_modif = x"0E")) else
         '1' when (cp_add = "11" and nAS ='0' and (adr_modif= x"09" or adr_modif = x"0D")) else
         '0';

--*** Selection de la direction pour le buffer
--*** 0 : ecriture et etat de repos
--*** 1 : lecture

nDATA_CS  <= not ( valid23 or valid32 ) ;
DATA_DIR <= nWRITE ;

--*** Attribution du Dtack

--*** signal de DATA ACKNOWLEDGE interne du decodeur
--*** actif si :
--***      - l'un des elements de la carte est selectionne
--***      - l'un des 2 DATA STROBE est actif

process(nRESET,CK16)
begin
  if nRESET = '0' then
    dtack_int <= ( others =>'0');
  elsif rising_edge (CK16) then
    if nds /= "11" then
      dtack_int(0) <= valid23 or valid32 ;
    else
      dtack_int(0) <= '0';
    end if;
    if nAS = '0' then
      dtack_int(2 downto 1 ) <= dtack_int(1 downto 0 );
    else
      dtack_int(2 downto 1 ) <= "00";
    end if;
  end if;
end process ;

nDTACK <= not dtack_int(2);

--***  Signal d'ecriture des differents registres internes , le signal
--*** est genere pour un usage interne .
process(nRESET, CK16)
begin
  if nRESET ='0' then
    ECRIT <= '0';
  elsif rising_edge (CK16) then
    if ( dtack_int ="001" and nWRITE ='0') then
      ECRIT <= '1';
    else
      ECRIT <= '0';
    end if;
  end if ;
end process ;

--*** Ecriture dans les registres par le bus VME
--Ecriture_registres:
process(nRESET, CK16)
begin
  if nRESET ='0' then
    int_reg <= ( others =>'0');
  elsif rising_edge(CK16) then
    if ECRIT ='1' then
      int_reg <= data;
    end if;
  end if ;
end process ;

INVERT <= int_reg(0);

data   <= int_reg when nWRITE ='1' and ( valid23 ='1' or valid32 ='1' ) else
          (others => 'Z');

-----------------------------------------------------------------
--          Internal MPS signal generation ( 120HZ, 30HZ )
-----------------------------------------------------------------

-- Clock Cycle generation

process (nRESET, CK16)
```

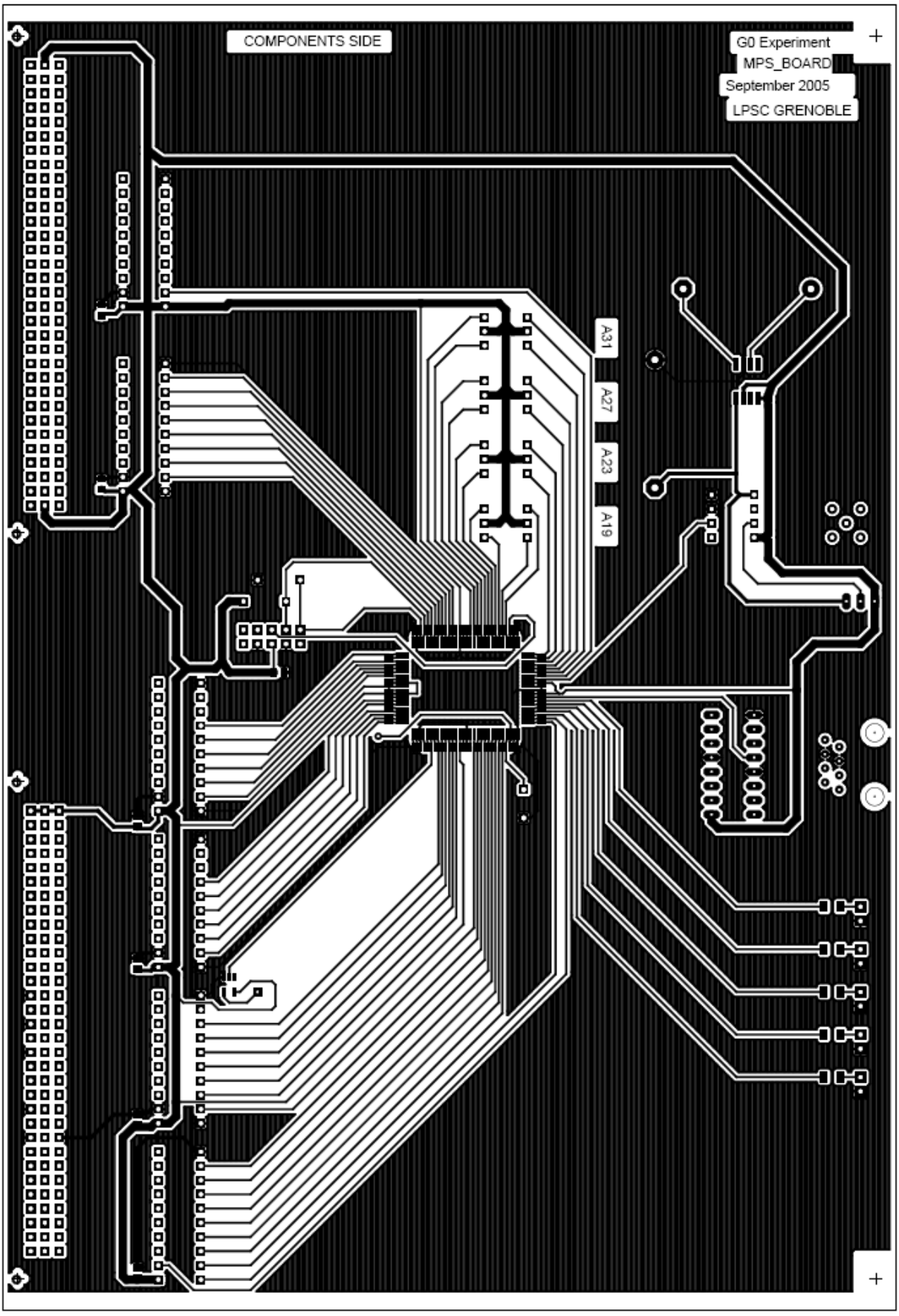
```
begin
  if nRESET ='0' then
    cpt_cycle   <= ( others =>'0');
    carry_cycle <= '0';
  elsif rising_edge(CK16) then
    if ECRIT ='1' then
      cpt_cycle   <= ( others =>'0');
      carry_cycle <= '0';
    else
      case int_reg(3) is
        when '0' => if cpt_cycle = BASE_CYCLE_30HZ then
                      cpt_cycle   <= ( others =>'0');
                      carry_cycle <= '1';
                    else
                      cpt_cycle <= cpt_cycle + 1;
                      carry_cycle <= '0';
                    end if;
        when '1' => if cpt_cycle = BASE_CYCLE_120HZ then
                      cpt_cycle   <= ( others =>'0');
                      carry_cycle <= '1';
                    else
                      cpt_cycle <= cpt_cycle + 1;
                      carry_cycle <= '0';
                    end if;
        when others => null;
      end case;
    end if;
  end if;
end process;

-- State Cycle generation

process (nRESET, CK16)
begin
  if nRESET ='0' then
    cpt_length   <= ( others =>'0');
    MPS_INT_GEN  <= '1';
  elsif rising_edge(CK16) then
    if ECRIT ='1' then
      MPS_INT_GEN <= '1';
      cpt_length  <= ( others =>'0');
    elsif carry_cycle ='1' then
      case int_reg(3) is
        when '0' => if cpt_length = x"000" then
                      cpt_length  <= cpt_length + 1;
                      MPS_INT_GEN <= '1';
                    elsif cpt_length = LENGTH_30HZ then
                      cpt_length <= ( others =>'0');
                    else
                      cpt_length  <= cpt_length + 1;
                      MPS_INT_GEN <= '0';
                    end if;
        when '1' => if cpt_length = x"000" then
                      cpt_length  <= cpt_length + 1;
                      MPS_INT_GEN <= '1';
                    elsif cpt_length = LENGTH_120HZ then
                      cpt_length <= ( others =>'0');
                    else
                      cpt_length  <= cpt_length + 1;
                      MPS_INT_GEN <= '0';
                    end if;
        when others => null;
      end case;
    end if;
  end if;
end process;


-----------------------------------------------------------------
--          MPS SIGNAL CONTROLER
-----------------------------------------------------------------

MPS_INT <= IN_422      when int_reg(4 downto 1) = "0001" else
           IN_LEMO     when int_reg(4 downto 1) = "0010" else
           MPS_INT_GEN when int_reg(4 downto 1) = "0100" else  -- mode 120 Hz
           MPS_INT_GEN when int_reg(4 downto 1) = "1000" else  -- mode 30 Hz
           '1';

MPS_OUT <= MPS_INT when INVERT ='0' else
           not MPS_INT ;


LED_IN_422   <= int_reg(1);
LED_IN_LEMO  <= int_reg(2);
LED_30HZ     <= int_reg(4);
LED_120HZ    <= int_reg(3);
LED_INVERT   <= int_reg(0);

end ;
```

Annex 5 : Board scheme

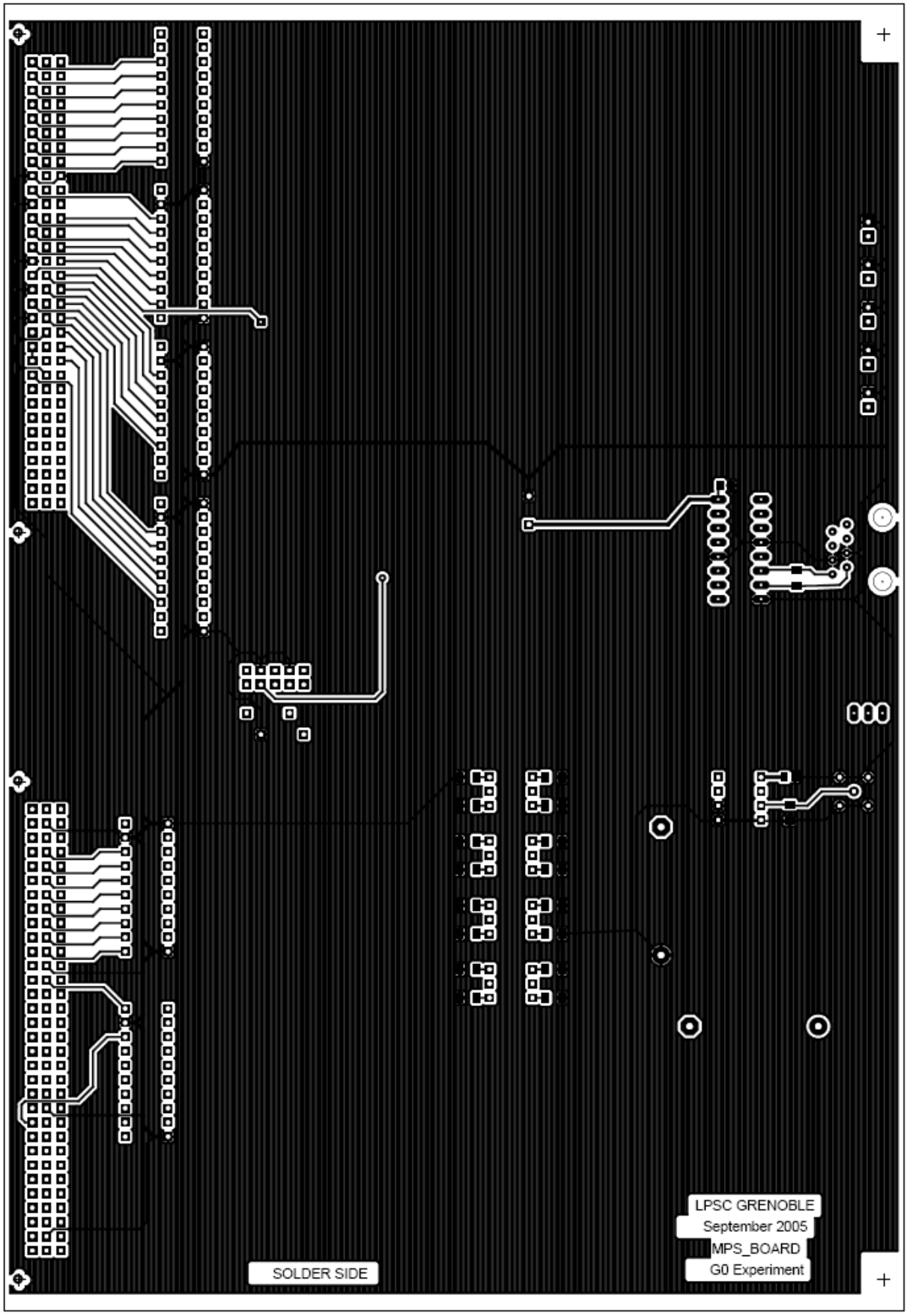



VCC
LM2662
C1
FC
CAP+
GND
CAP-
V+
OSC
LV
OUT
-5V
C2
GND
R11
TRIM_EU-B64Y
U2
VCC
-5V
IN_LEMO
LEM1
LE
R27
50
CD10
GND
GND
LED_IN_422
R3
IN_422
LED_IN_LEMO
R4
IN_LEMO
LED_30HZ
R5
30HZ
LED_120HZ
R6
120HZ
LED_INVERT
R7
INVERT
GND
IC8
AM26C32
GND
IN_422
J2
R8
CD9
TITLE: MPS_BOARD1
Document Number:
REV:
Date: 29/09/2005 15:54:10
Sheet: 2/2

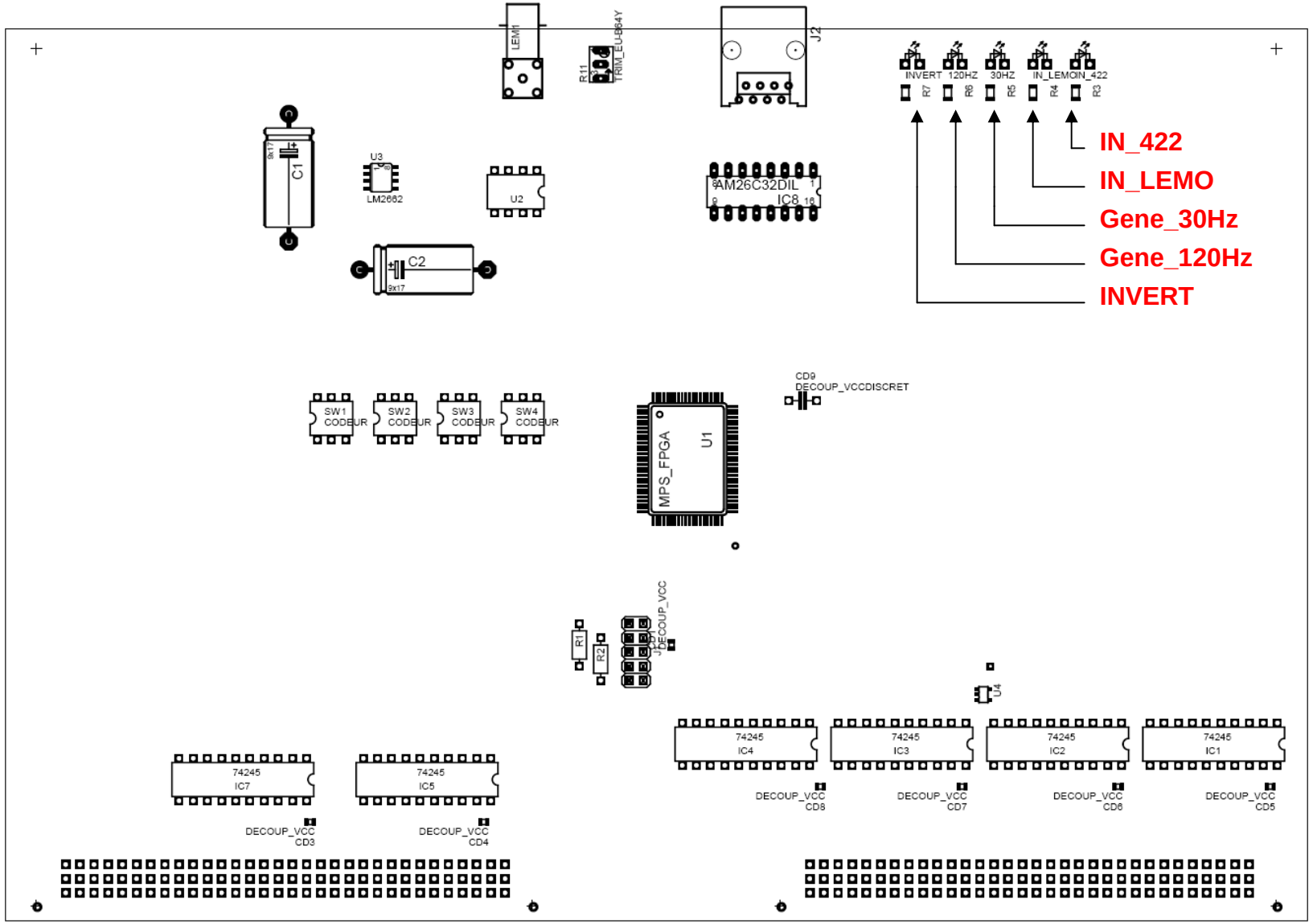
Annex 6 : Top Layer view



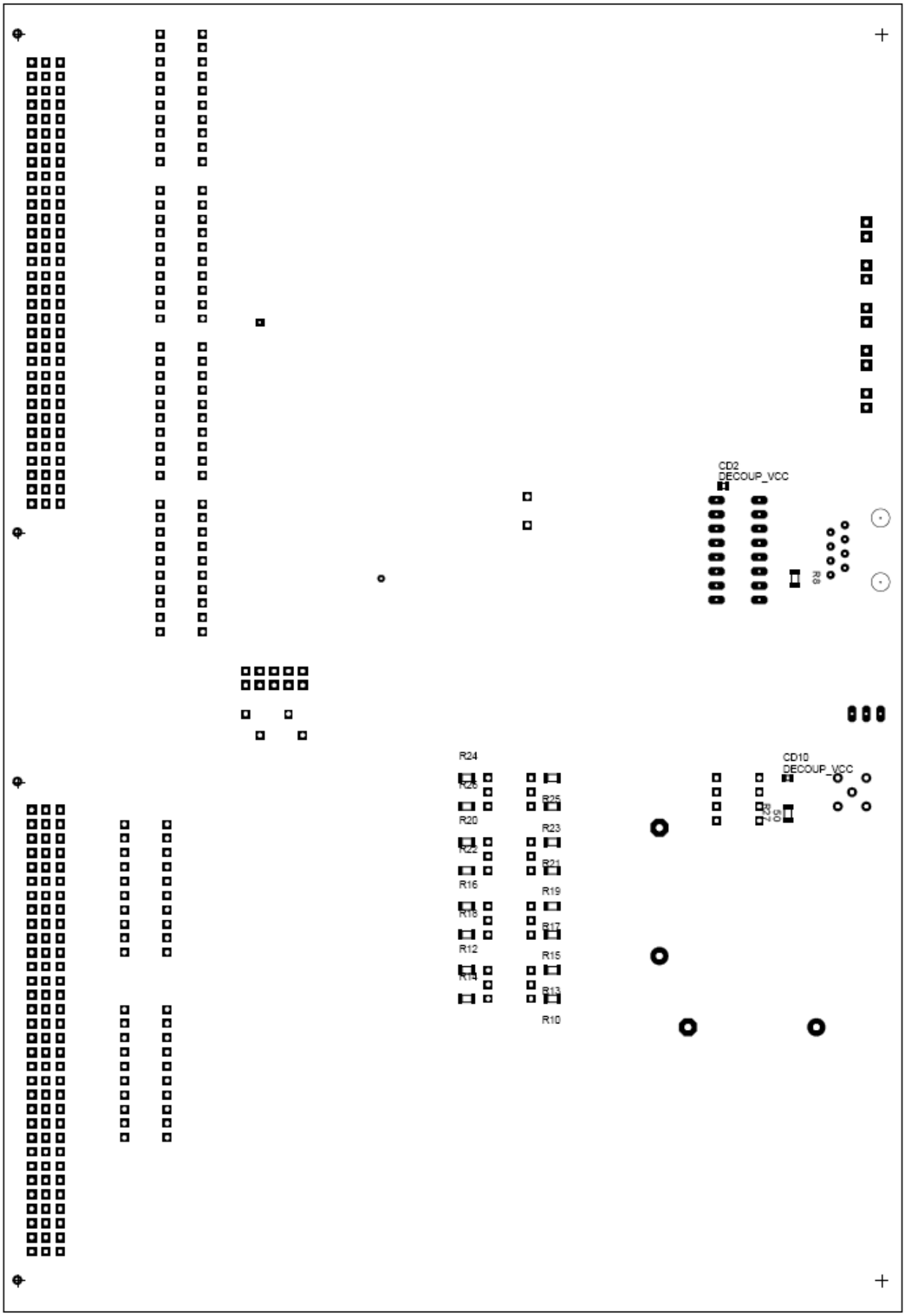
Annex 7 : bottom Layer view



Annex 8 : Top component view



Annex 9 : Bottom layer view



Annex 10 : Switched Capacitor Voltage Converter

National Semiconductor

January 1999

LM2662/LM2663
Switched Capacitor Voltage Converter

General Description

The LM2662/LM2663 CMOS charge-pump voltage converter inverts a positive voltage in the range of 1.5V to 5.5V to the corresponding negative voltage. The LM2662/LM2663 uses two low cost capacitors to provide 200 mA of output current without the cost, size, and EMI related to inductor based converters. With an operating current of only 300 µA and operating efficiency greater than 90% at most loads, the LM2662/LM2663 provides ideal performance for battery powered systems. The LM2662/LM2663 may also be used as a positive voltage doubler.

The oscillator frequency can be lowered by adding an external capacitor to the OSC pin. Also, the OSC pin may be used to drive the LM2662/LM2663 with an external clock. For LM2662, a frequency control (FC) pin selects the oscillator frequency of 20 kHz or 150 kHz. For LM2663, an external shutdown (SD) pin replaces the FC pin. The SD pin can be used to disable the device and reduce the quiescent current to 10 µA. The oscillator frequency for LM2663 is 150 kHz.

Features

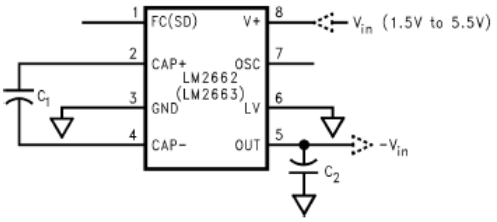
- Inverts or doubles input supply voltage
- Narrow SO-8 Package
- 3.5Ω typical output resistance
- 86% typical conversion efficiency at 200 mA
- (LM2662) selectable oscillator frequency: 20 kHz/150 kHz
- (LM2663) low current shutdown mode

Applications

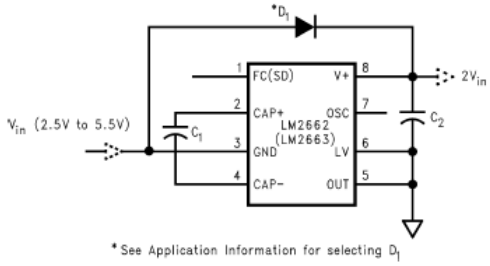
- Laptop computers
- Cellular phones
- Medical instruments
- Operational amplifier power supplies
- Interface power supplies
- Handheld instruments

Basic Application Circuits

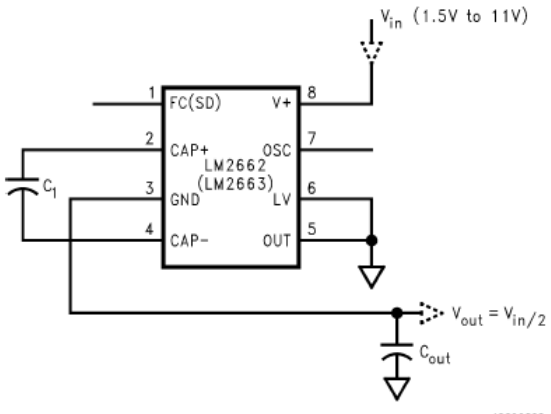
Voltage Inverter



Positive Voltage Doubler



Splitting V_{IN} in Half



 DS100003 www.national.com

TABLE OF CONTENTS